

# Overview

- What is OSM?
- Why does Kosovo need maps?
- Why does OSM make sense for Kosovo?
- What we have done so far?
- Why you should help with the maps?
- How you can help ?

# OSM = GIS

- Geographic (Geometric)
  - Points (Shops)
  - Lines (Streets)
  - Areas (Buildings, Football Fields, Schools)
  - Textures/Heights (Mountains)
- Information
  - Bits and Byte (XML) Compression (bzip2) and Data Transfer (HTTP)
- Systems (Programs)
  - Input/Processing/Output

# Database

- Values of types of data (Integers, Strings, Decimal Numbers, Coordinates etc)
- Rows of Data which represent objects
  - Hotel Victory;Mother Theresa str.;10;Prishtina
- Columns of Data
  - Name;Street;HouseNumber;City

# Tables of Data

| Name (String) | Street(String)    | House Number<br>(Integer) | City (String) |
|---------------|-------------------|---------------------------|---------------|
| FEIK          | Agim<br>Ramadani  | 1                         | Prishtina     |
| Hotel Victory | Mother<br>Theresa | 2                         | Prishtina     |

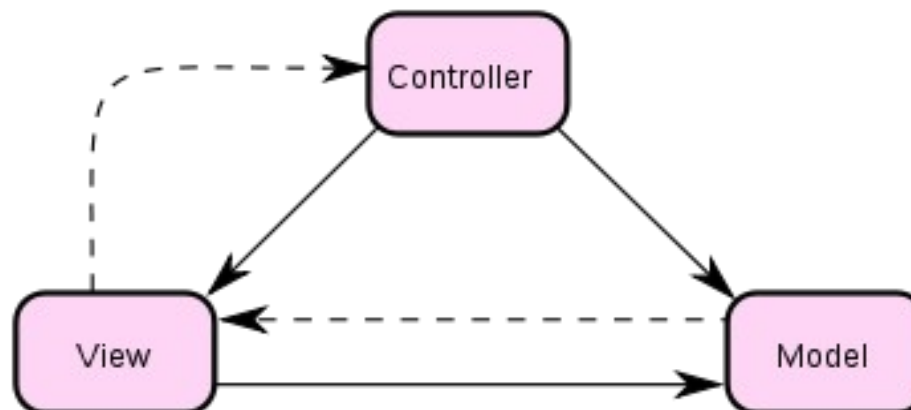


# Postgres/POSTGIS

- Database Queries = Asking the database questions and getting answers
- SQL=Standardized Query Language
- Postgres is an SQL database like MySQL with many plugins, for example the GIS extensions
- Example: `select c1.cntry_name as Kosova_Neighbors from country c1, country c2 where touches(c1.the_geom,c2.the_geom)='T' and c2.cntry_name='Kosova';`

# Ruby on Rails

- Ruby is a scripting language like php or perl
- Ruby on Rails is a popular **MVC** Model View Controller web framework
  - Model = Database Model
  - View = Html/ PNG Display
  - Controller = User actions that change the model (Programs that react)



# OSM

## Open

- Anyone can help and change the map
- Not a company , not a Government agency
- Don't need a special permission
- Copyleft / Copyright License not a Contract

- Street

- Streets and Buildings, Tourism Data, Shops, Hotels etc.

- Map

- 2d Picture of Geographical Data

# openstreetmap.org

- Main webserver is Ruby application with a postgres database for viewing and editing maps
- Many applications running on many servers
  - Database server – postgres with gis extensions
  - Rendering creating of PNG Tiles – mapnik
  - Potlatch Editor / JOSM
  - Planet file (database snapshots) dumps
    - Name Server <http://nominatim.openstreetmap.org/>
- Community with blogs, messages, irc, mailing lists forums and wiki

# Community

Mailing List

<http://lists.openstreetmap.org/listinfo/talk>

Forum:

<http://forum.openstreetmap.org/>

Wiki:[wiki.openstreetmap.org](http://wiki.openstreetmap.org)

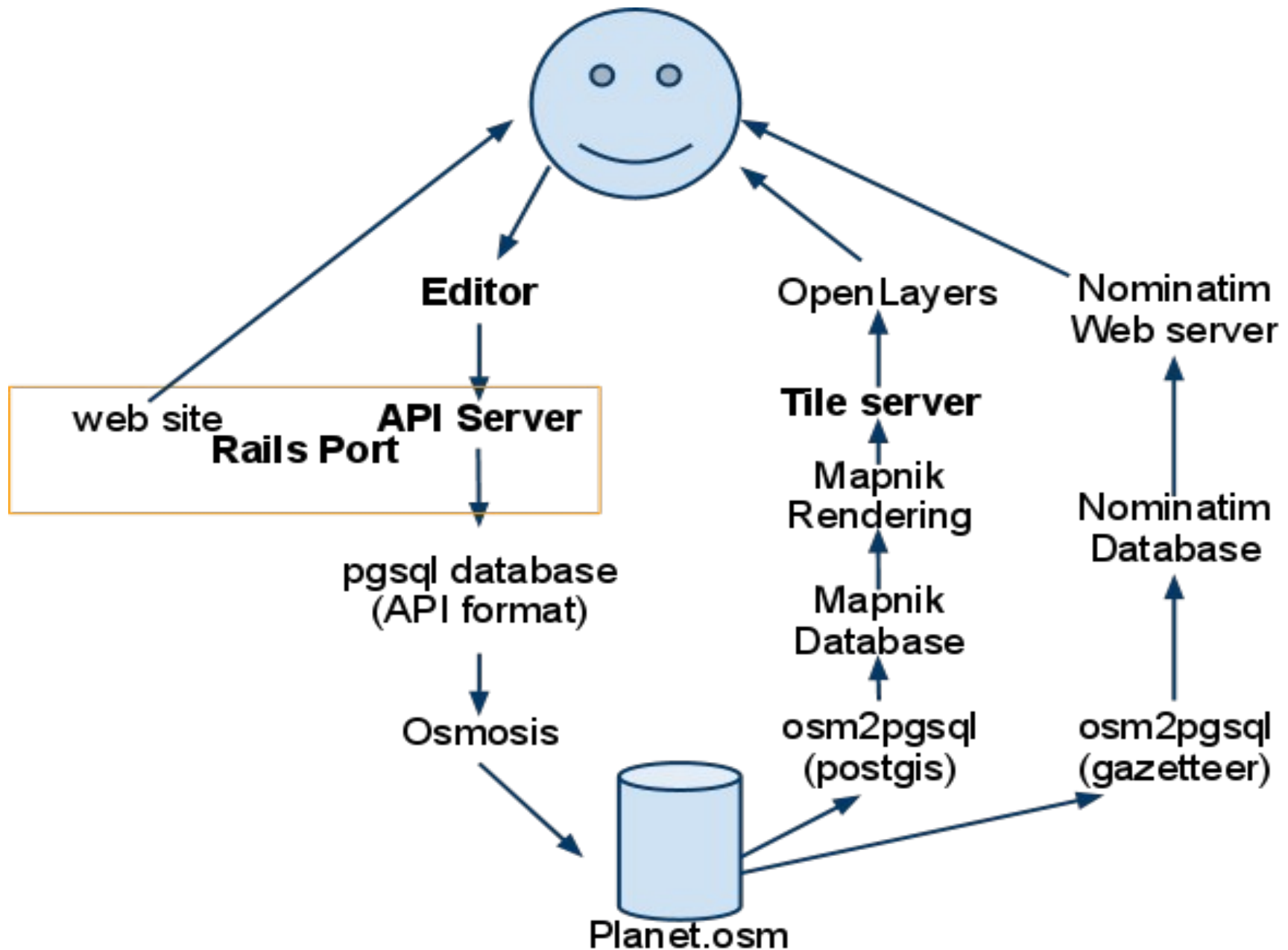
IRC Chat:irc.oftc.net #osm #osm-dev #osm-de

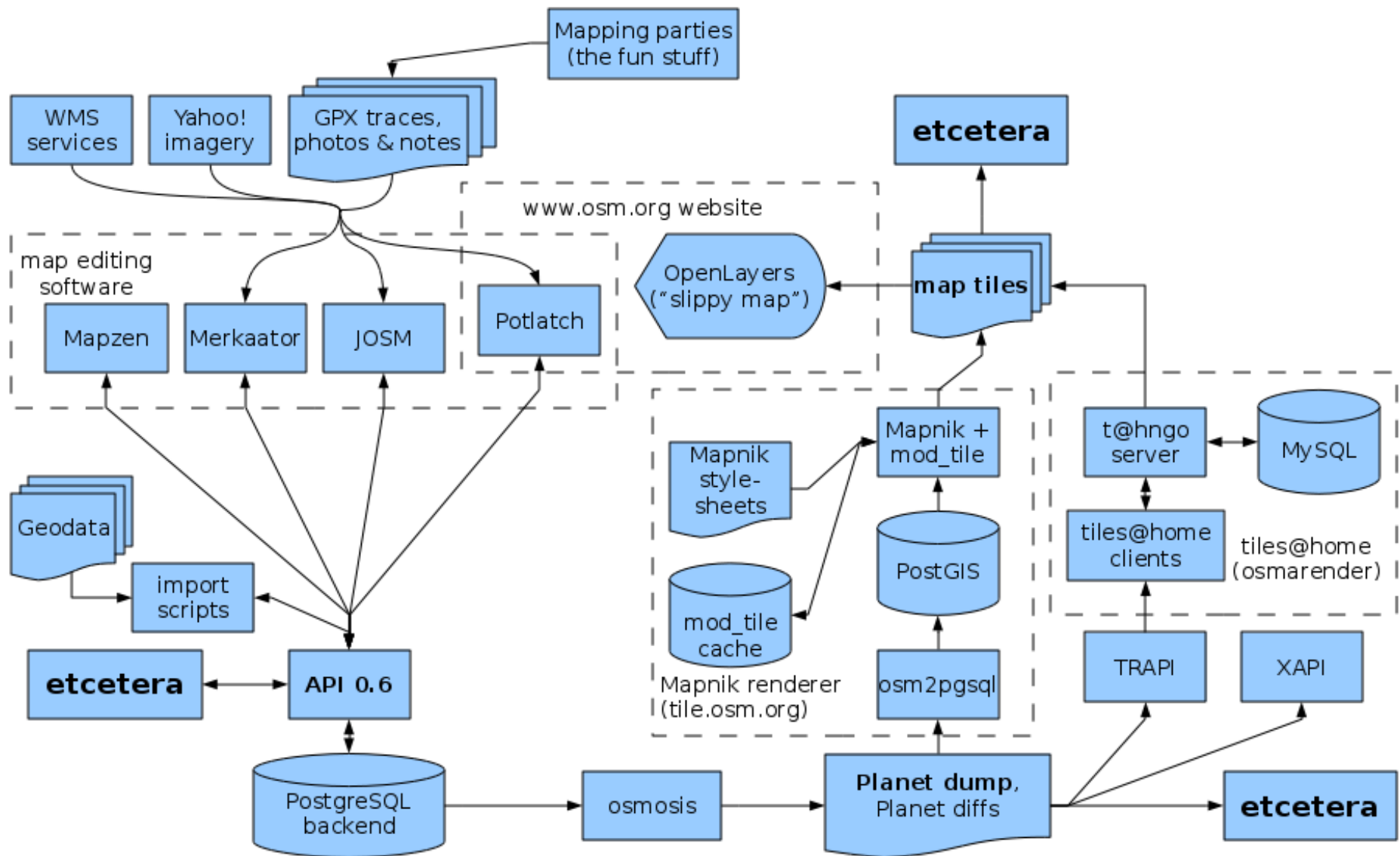
# Component Overview

[http://wiki.openstreetmap.org/wiki/Component\\_overview](http://wiki.openstreetmap.org/wiki/Component_overview)

- [2 Database](#)
- [3 API](#)
- [4 OSM Front End](#)
  - [4.1 Slippy Map](#)
  - [4.2 Potlatch](#)
- [5 Tiles and tile rendering](#)
  - [5.1 Mapnik](#)
  - [5.2 Tiles@home](#)
- [6 Renderers](#)
- [7 Editors](#)

# OpenStreetMap dataflow





source :

[http://wiki.openstreetmap.org/wiki/File:OSM\\_Components.png](http://wiki.openstreetmap.org/wiki/File:OSM_Components.png)



# Why Kosovo needs maps

- Kosovo is a beautiful country with many hidden treasures, they are hidden because no one can find them easily!
- Tourism will be promoted by historical monuments , restaurants and hotels being accessible
- Maps are needed for mobile gps devices, usable by tourists, taxis and people
- Investors want to know where property is available to purchase











# Missing Signs and Information

- Missing road signs , many road signs and directional signs are missing.  
<http://www.flickr.com/photos/zanthia/3881839415/#/>
- Missing house numbers, many houses are not clearly numbered.
- Unclear road names, old and new names. No addresses on houses, on streets, or business cards or webpages.
- Road construction information is needed to be updated in a timely fashion.





TURIZËM RURAL  
SEOSKI TURIZAM  
RURAL TOURISM

Qyteti i vjetër i Kalasë  
Stari grad u Tvrđjavi  
Old town at the Fortress

RURAL  
TOURISM  
RURAL TOURISM

Fshati BUNJAK  
Selo BUNJAK  
Village BUNJAK

TURIZËM RURAL  
RURAL TOURISM

Ministria e Kulturës dhe Turizmit  
Ministry of Culture and Tourism

PAINT. BALL →



Prishtinë  
Pristina  
Priština  
Gjakovë  
Djakovica  
Yakova

Qendër  
Centar  
Merkez

# Tourism is promoted with maps

- Tourist and local visitors need maps
- to plan their visit and locate important resources like food and places to sleep.
- Bus stations, Bus routes and timetables are badly needed online for kosovo!
- Roads conditions need to be clearly marked on the maps.
- Locations and contact details are needed for all restaurants and hotels

<http://wikitravel.org/en/Pristina>



# Schools need Maps

- Children to find playgrounds and Parks
- For Planning Excursions with School
- Geography Lessons with Up to data data
- For students in new cities (Coming to Prishtina)

# Maps provide important information

## Infrastructure needed for commerce

- Market information and stimulate competition and investment
- Train, Boat, Bus and Air schedules must be clear and accessible and timely available
- Travel Visa, Insurance, Customs and Toll information must also be available
- The raw data is very valuable not just the printed form of a map

# Why Kosovo needs OSM

- Localization in both Gheg and Standard Albanian possible and underway
- People can use the names they want : Republic of Kosova versus Kosovo Greater Serbia
- The data is owned by the people, not a company or government agency, freedom of information
- Simulation of the economy, opening of new markets in derived works and innovative localized applications marketed by local shops, internet savvy users and even local software development teams

# Many Names of places and languages

- Old and New Names
- Variant names
- English, Albanian, Serbian and Turkish names
- All supported by osm

# Getting an account

- Anyone can register and get an account

<https://www.openstreetmap.org/user/new>

- All changes are saved against a person, in a changeset
- All vandalism is traceable

<http://www.openstreetmap.org/browse/node/856218729>

- Changes appear after a certain time on the main page



# Osm wiki

<http://wiki.openstreetmap.org>

- Make your Home page
- Project Pages for Places and groups
- Documentation of tools and services
- Documentation of file formats
- Keys and Tags

# Archive.org

- Unlimited storage of data forever
- Creative commons and public domain
- Somewhat slow access time
- FTP upload
- Managing of videos and audio files as well

# Licensing issues

- Cannot use maps that are not public domain/creative commons without permission
- Cannot use Kadaster Data and orthophotos
- Can use city maps with permission
- Cannot use google or other maps
- Cannot use Commercial maps

# Collecting of raw data

- Gps data
- Photomapping
- Walking papers
- Recitfied maps
- Adding from memory
- Lists of streetnames







# Albania

Importing of street data

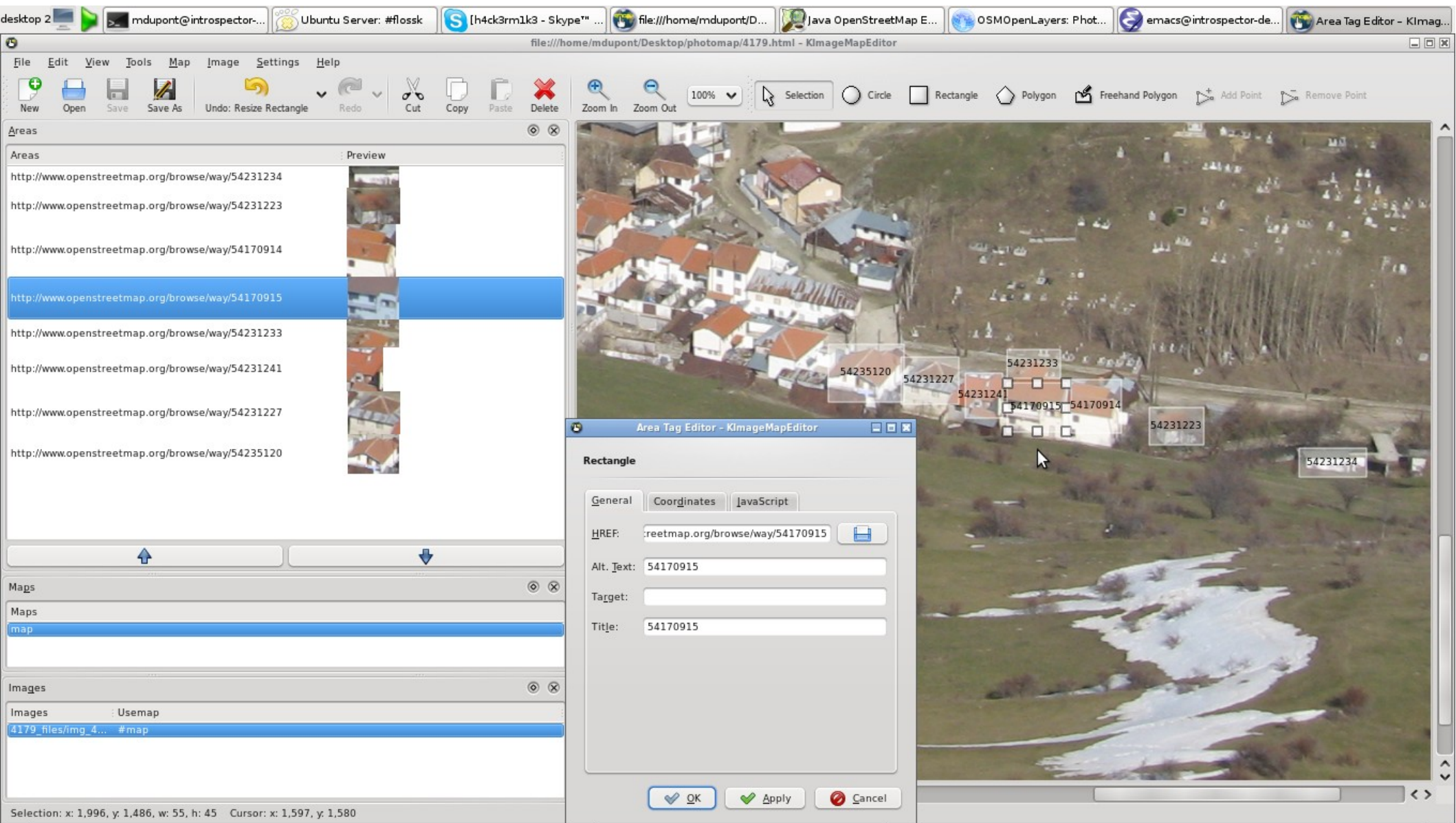
GNS names

Bunkertrails

Existing Data

Agim Kompani

TPGINC.net





# warper.geothings.net

- You can upload photos and georeference them
- Usage of gdalwarp and mapserv internally
- Open source application

# Potlatch editor

- Flash Editor for the OSM webpage
- Can be installed on your own server, cannot talk to remote osm server

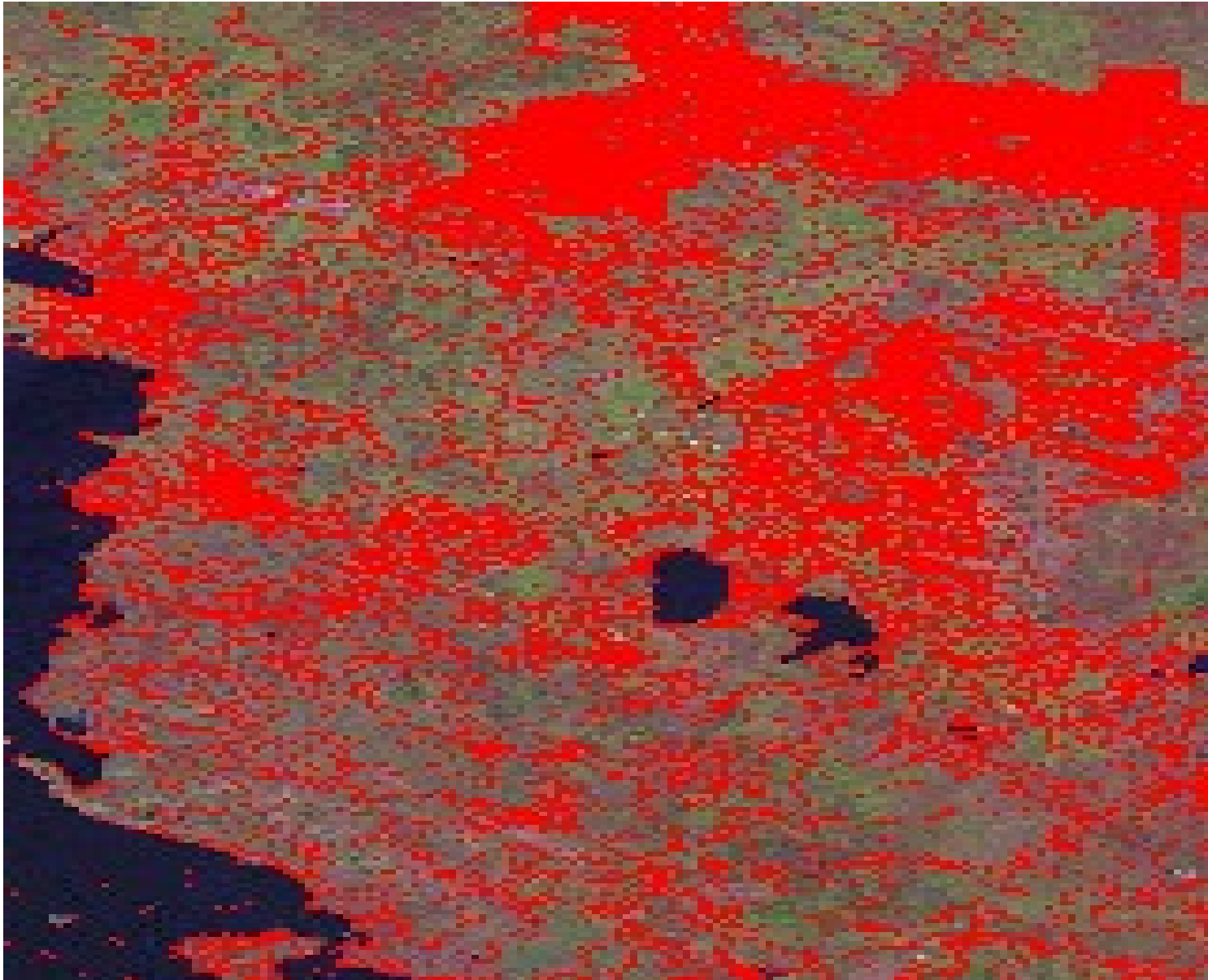
# josh

- Java Desktop editor with many plugins and features

# History and development in kosovo

- Prehistory
- Logistics plus meeting
- Call for donations of data
- Various Imports
- Projects
  - Cities
  - Logistic Plus
  - UN Habitat
  - degradimi
  - Brod

# View of history of changes



# Comparison 2 googlemaps

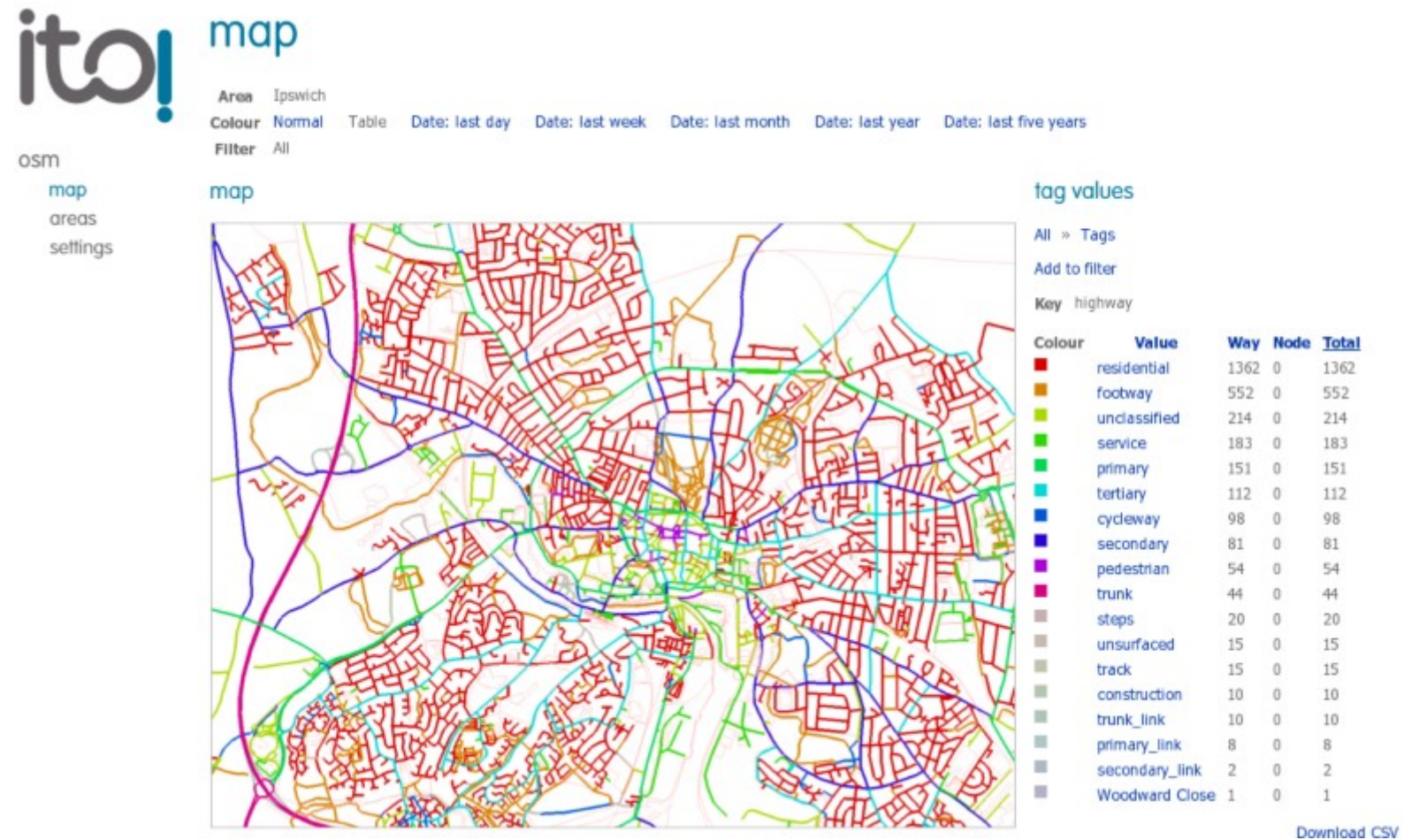
- Kosova not Serbia
- License: With osm you own all your data, with google they own it.
- Routing and searching
- Usage on GPS devices, offline access
- Static Bitmaps cannot be used
- Usage for videos and in television
- Commercial Usage, name lookup



# ITOWorld

- Nice online tool (osm mapper [www.itoworld.com](http://www.itoworld.com) )
- Not FLOSS
- History and user view
- commercial company
- (1 of many) that use of OSM data and provide it in creative and unexpected ways to the end-user (just provide a unique 'window' into the database)

# ITO World webpage



# Wikimapia

- <http://wikimapia.org/#lat=42.6530927&lon=21.165143>
- Cannot include the picture in my slide because of license issues.
- Cannot use data in openstreetmap

# Routing

- <http://wiki.openstreetmap.org/wiki/Routing>
- Some of the routing applications
- Examples using kosovo
- pr-pz:  
<http://openrouteservice.org/index.php?start=21.1620686>
- In city  
<http://openrouteservice.org/index.php?start=21.166875,4>

Comparision to other countries

# Business Usages

- Creating maps for restaurants/hotels
- Validating addresses/geocoding
- Usage in press and media
- Storing/sharing datasets
- Custom solutions
- placenames

# Mobile Applications

- Iphone
- Mapdroid
- Nokia whereami
- Garmin devices

# World files

Daily dump of entire world

Split into countries

Can process with osmosis

Usable with simple tools



# Embedding maps

- Png and html export
- Link to nodes/links to positions

# Printing tools

- Walking papers
- Townguide
- Osmomatic
- Osmrender
- See wiki page
- [http://wiki.openstreetmap.org/wiki/OSM\\_on\\_Paper](http://wiki.openstreetmap.org/wiki/OSM_on_Paper)

# nomatim

- Name lookup
- Needs own database import
- Can find things near others
- Can be used to lookup addresses

# Osm data structures

- Users
  - Changesets
- World
  - Bounding box
- Points
- Ways
- Relations

# Conversion tools

- SHP2OSM
- Osmosis
- JOSM
- gdal/ogr
- Hand made perl scripts

# Qgis

<http://www.qgis.org/>

# Gps tracking

The screenshot shows the PicLayer WMS software interface. The top menu bar includes 'Datei', 'Bearbeiten', 'Ansicht', 'Werkzeuge', 'Vorlagen', 'PicLayer', 'WMS', 'Audio', and 'Hilfe'. The toolbar contains various icons for file operations, navigation, and editing. The main map area displays a street map with a scale bar indicating 0 to 500.4 meters. The right sidebar contains three panels: 'Ebenen' (Layers) with 'Datenebene 1' (unchecked), 'GPX Heruntergeladene GPS-Daten' (checked), 'Ergebnisse der Datenprüfung' (checked), and 'Karte' (unchecked); 'Eigenschaften / Mitgliedschaften' (Properties / Memberships) with a message 'Bitte wählen Sie die Objekte, deren Eigenschaften Sie ändern wollen.' and buttons 'Hinzufüg...', 'Bearbeit...', and 'Entfernen'; and 'Auswahl' (Selection) with a list of selected objects and buttons 'Auswahl', 'Suche', 'Aus...', 'Prüf...', 'Repa...', and 'Igno...'. The bottom status bar shows coordinates '42.3821233' and '20.4515060', a cursor icon, and the text '(kein Objekt)'. The bottom right corner contains the text 'Auswahl hinzufügen. Strg: Auswahl umschalten. Umschalt-Strg: drehen oder Auswahl ändern.'

data col... present... [andrea... #flossk Java-Op... Musik - ... Soul of ... Fr, 24. Sep, 23:31 andreas

Datei Bearbeiten Ansicht Werkzeuge Vorlagen PicLayer WMS Audio Hilfe

0 500.4 m

Ebenen

- ☒ Datenebene 1
- ☒ GPX Heruntergeladene GPS-Daten
- ☒ Ergebnisse der Datenprüfung
- ☒ Karte

Eigenschaften / Mitgliedschaften

Bitte wählen Sie die Objekte, deren Eigenschaften Sie ändern wollen.

+ Hinzufüg... Bearbeit... Entfernen

Auswahl

Auswahl Suche

Autoren

☒ Ergebnisse der Datenprüfung

Aus... Prüf... Repa... Igno...

42.3821233 20.4515060 (kein Objekt)

Auswahl hinzufügen. Strg: Auswahl umschalten. Umschalt-Strg: drehen oder Auswahl ändern.



# Jugoslav Military Maps

[http://wiki.openstreetmap.org/wiki/Kosovo\\_iMMAp\\_Import](http://wiki.openstreetmap.org/wiki/Kosovo_iMMAp_Import)

# Projections

[http://docs.openlayers.org/library/spherical\\_mercator.html](http://docs.openlayers.org/library/spherical_mercator.html)

Projections in GIS are commonly referred to by their “EPSG” codes, identifiers managed by the European Petroleum Survey Group. One common identifier is “EPSG:4326”, which describes maps where latitude and longitude are treated as X/Y values. Spherical Mercator has an official designation of EPSG:3785. However, before this was established, a large amount of software used the identifier EPSG:900913. This is an unofficial code, but is still the commonly used code in OpenLayers. Any time you see the string “EPSG:4326”, you can assume it describes latitude/longitude coordinates. Any time you see the string “EPSG:900913”, it will be describing coordinates in meters in x/y.

# Kosovo projections

- 1. Yugoslavia used Gauss Kruger with Bessel Spheroid (not Pulkovo, not Krassovsky etc...) This is a localized UTM Map Projection!
- 2. UN started in 1999 with UTM projection and WGS84 and more or less all other organizations (even some government) followed that path
- 3. Kosova Cadastral Agency adopted a Euref (EU compliance) system in Kosova context with in principle they replaced Bessel Spheroid with WGS84 Datum and are now calling it KosovaRef!.

# KPA Property agency

<http://www.kpaonline.org/>

cs2cs -E +proj=utm  
+zone=34T +units=m  
+proj=tmerc -f "%.7f"

```
ogr2ogr -s_srs EPSG:2641 -t_srs EPSG:4326  
test3.shp
```

```
pulkovo/dragash_pulkovo.shp
```

<http://osmopenlayers.blogspot.com/2010/02/please-help->

SRC Pulkovo 1995 / 3-degree Gauss-Kruger  
zone 7:

<http://spatialreference.org/ref/epsg/2641/>

Not Pulkovo 1942 / Gauss-Kruger CM 39E :

<http://spatialreference.org/ref/epsg/2497>

Dest:

<http://spatialreference.org/ref/epsg/4326/>

2641test.shp (extension is shp)

File contains 1 layers

Detected projection metadata:

PROJCS["Transverse\_Mercator",

GEOGCS["Krassovsky, 1942",

DATUM["unknown",

SPHEROID["krass",6378245,298.3]],

PRIMEM["Greenwich",0],

UNIT["Degree",0.017453292519943295]],

PROJECTION["Transverse\_Mercator"],

PARAMETER["latitude\_of\_origin",0],

PARAMETER["central\_meridian",21],

# UTM Coordinates

[http://en.wikipedia.org/wiki/Easting\\_and\\_northing](http://en.wikipedia.org/wiki/Easting_and_northing)

The terms **easting** and **northing** are [geographic Cartesian coordinates](#) for a point. Easting refers to the [eastward](#)-measured distance (or the x-[coordinate](#)), while northing refers to the [northward](#)-measured distance (or the y-coordinate). The [orthogonal](#) coordinate pair are commonly measured in [metres](#) from a horizontal [datum](#).

# Proj4 projection

CS2CS

[http://proj.maptools.org/man\\_cs2cs.html](http://proj.maptools.org/man_cs2cs.html)

building proj with c++

<http://github.com/h4ck3rm1k3/ProjC-->



# Example Projection

Examples :Lon :8.6511725 Lat: 50.163021

-->x:475085.2164415381 y:5556814.616540354 Zone :32N

([epsg/32632](http://epsg.org/32632))

<http://home.hiwaay.net/~taylorc/toolbox/geography/geoutm.htm>

↓

<http://spatialreference.org/ref/epsg/32632/>

+proj=utm +zone=32 +ellps=WGS84 +datum=WGS84

+units=m +no\_defs to use , append +proj=tmerc -f

"%.7f" cs2cs +proj=utm +zone=32 +ellps=WGS84

+datum=WGS84 +units=m +no\_defs +proj=tmerc -f "%.7f"

475085.2164415381 5556814.6165403548

8.6511725 50.1630210 0.00000000

# UTM Zones

[http://en.wikipedia.org/wiki/Universal\\_Transverse\\_Mercator\\_coordinate\\_system](http://en.wikipedia.org/wiki/Universal_Transverse_Mercator_coordinate_system)

The UTM system divides the surface of Earth between [80°S](#) and [84°N latitude](#) into 60 zones, each 6° of longitude in width and centered over a meridian of longitude. Zone 1 is bounded by longitude 180° to 174° W and is centered on the 177th West meridian. Zone numbering increases in an easterly direction.

# geonames

<http://www.geonames.org/>

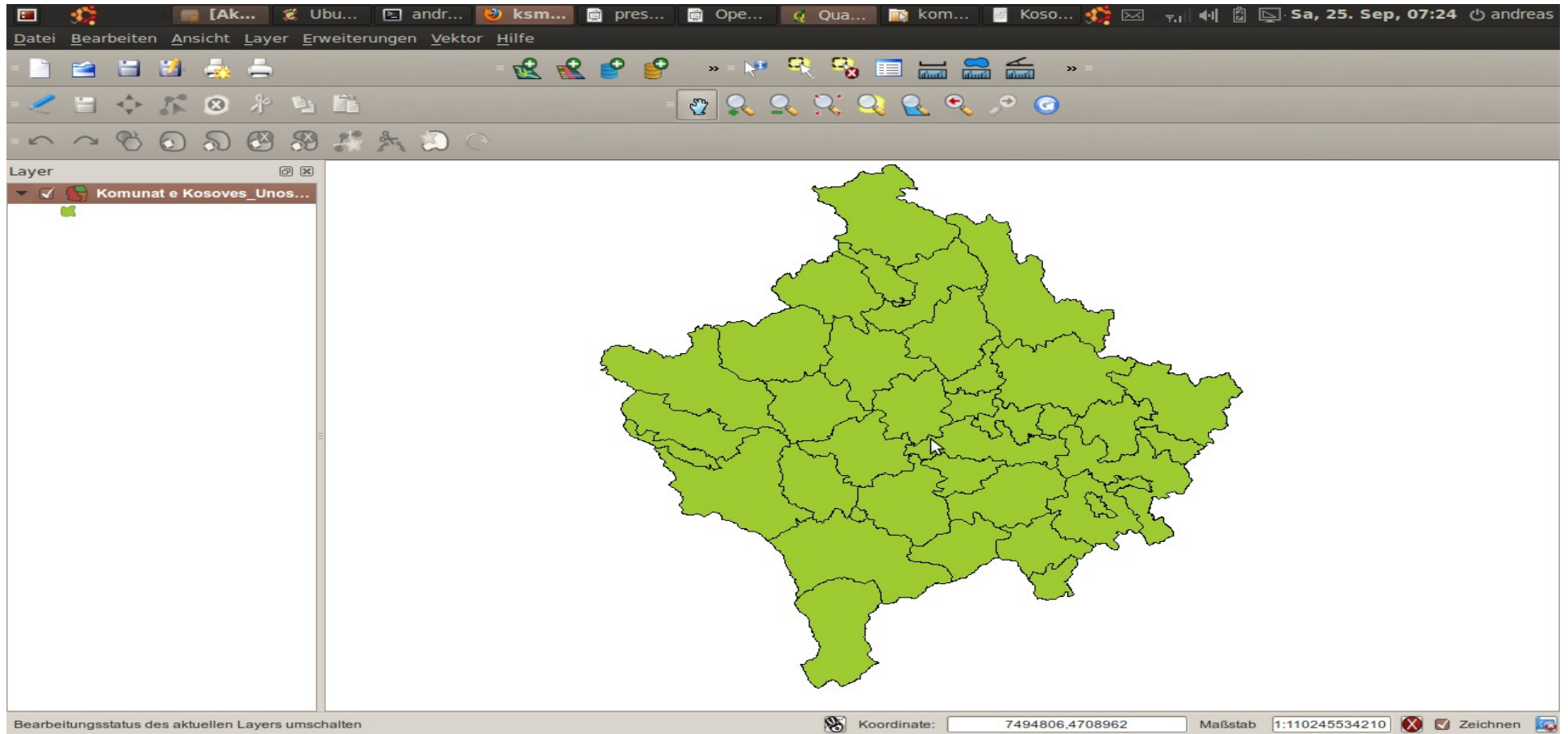
The GeoNames geographical database covers all countries and contains over eight million placenames that are available for download free of charge.

# National Geospatial-Intelligence Agency's names

National Geospatial-Intelligence Agency's (NGA) and the U.S.  
Board on Geographic Names (most names except US and  
CA)

<https://www1.nga.mil/ProductsServices/GeographicNames/Pages/default.aspx>

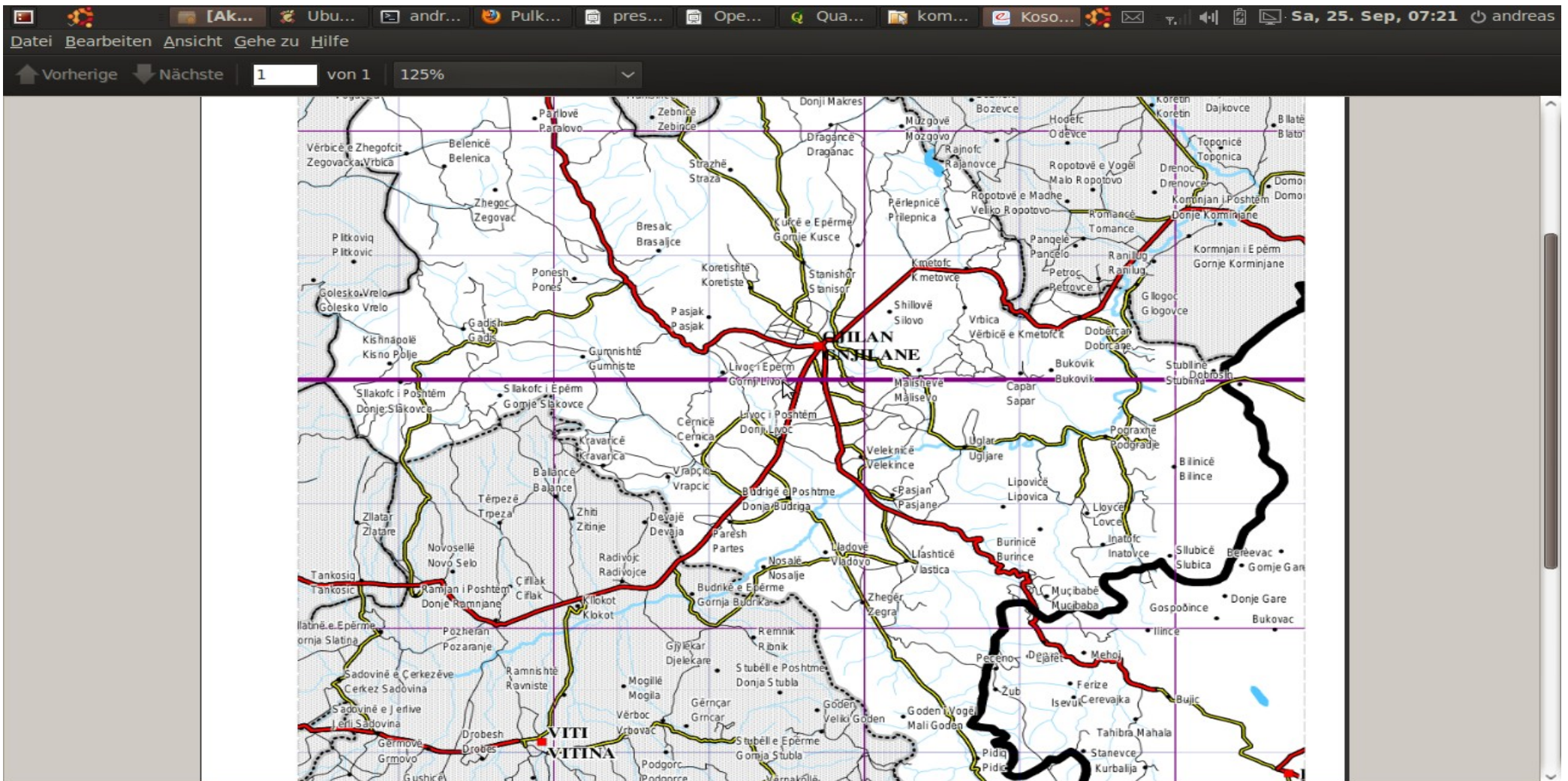
# State boundaries



<https://docs.google.com/leaf?id=0B4QovvVR7xyxYWZlZ>

Shape file for state boundaries

# UNDP Maps





# Russian Maps

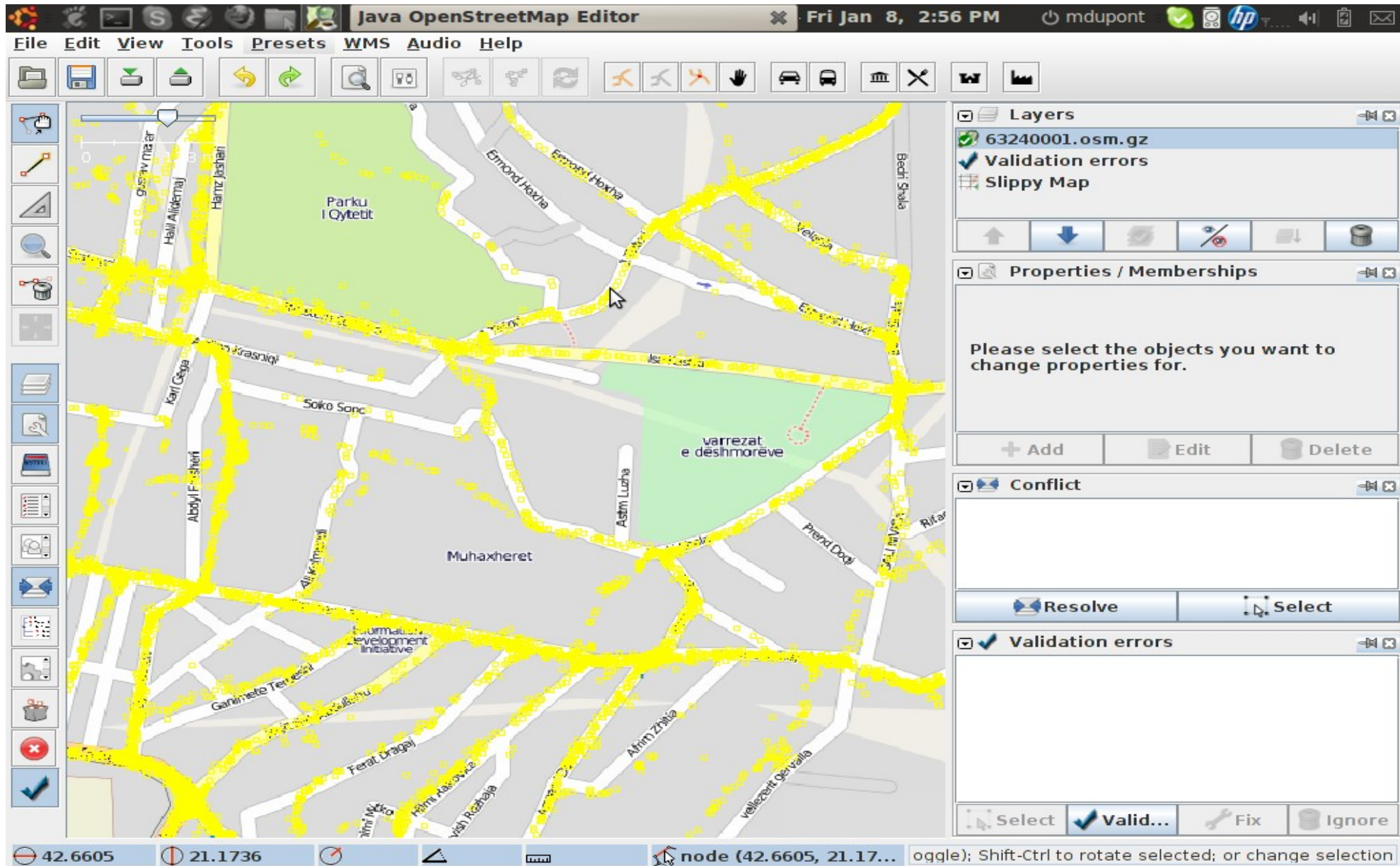


# Orthophotos – non free

[http://farm4.static.flickr.com/3311/3620710100\\_aebcf](http://farm4.static.flickr.com/3311/3620710100_aebcf)



# Logistics Plus



# degradimi

[http://degradimi.info/Faqja\\_Kryesore](http://degradimi.info/Faqja_Kryesore)

# DXF Prishtina

[http://wiki.openstreetmap.org/wiki/Kosovo\\_Prishtina\\_Streets\\_Tracing](http://wiki.openstreetmap.org/wiki/Kosovo_Prishtina_Streets_Tracing)

Here are the latest OSM files, split into  
10,000 point blocks.

<http://www.archive.org/details/PrishtinaStreetsOSM>

The original dxf file is here :

<http://www.archive.org/details/PrishtinaStreets>

the shifted one from Andrew is here :

<http://www.archive.org/details/PrishtinaOpensourceMap>

See my blogpost :

<http://osmopenlayers.blogspot.com/2010/09/mensa-traced-from-cad-files.html>

# Un habitat

Using of OSM data

Using of QGIS

Cooperation with FLOSSK

Mapping in Brod













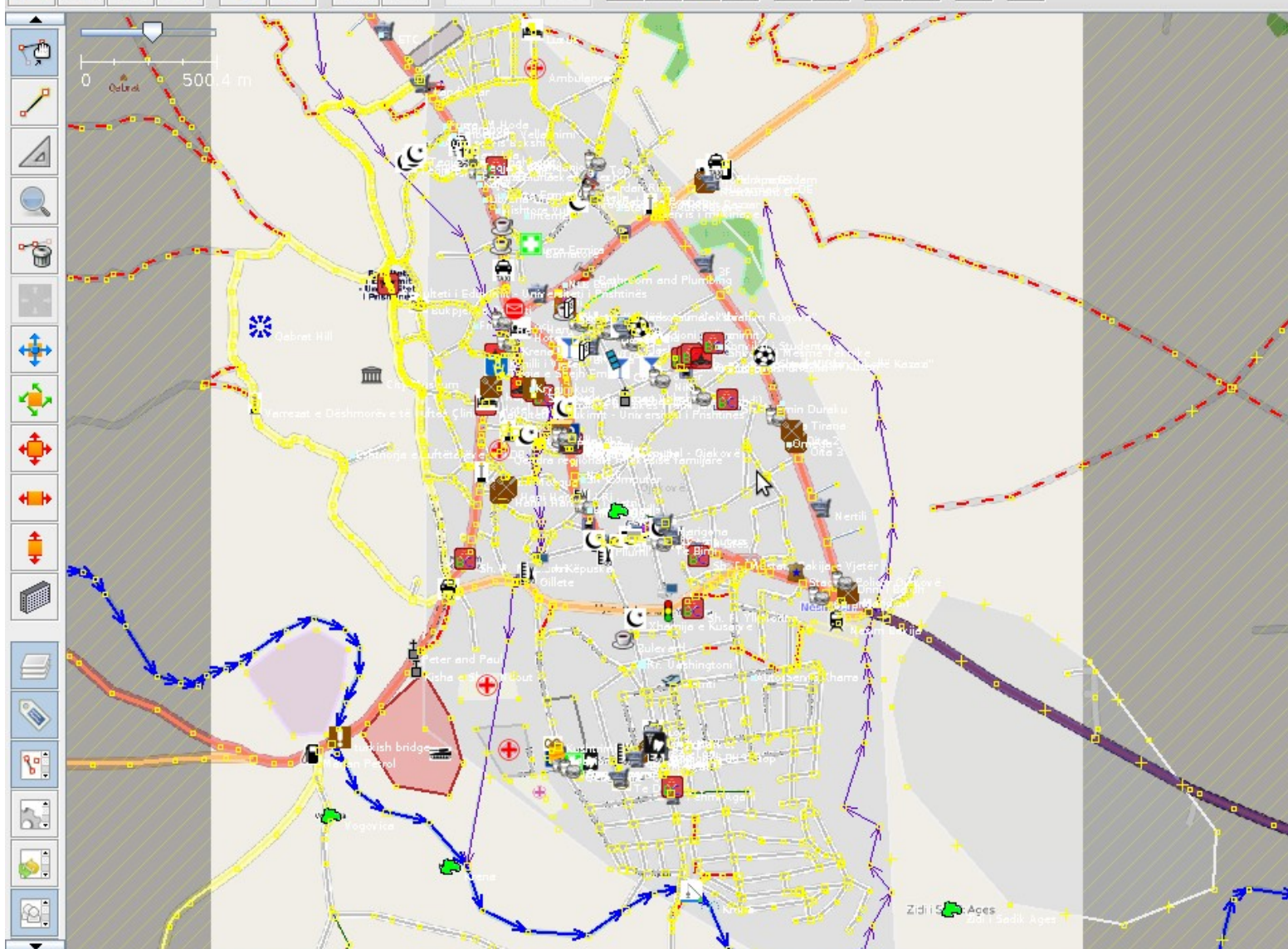






# Prishtina

- Jugoslav maps
- Official city maps
- CAD Files
- Yahoo sat
- GPS devices
- [http://wiki.openstreetmap.org/wiki/WikiProject\\_Prishtina](http://wiki.openstreetmap.org/wiki/WikiProject_Prishtina)



☒ Ebenen  
☒ Datenebene 1  
☒ Ergebnisse der Datenprüfung  
☐ Karte

☒ Eigenschaften / Mitgliedschaften  

Bitte wählen Sie die Objekte, deren Eigenschaften Sie ändern wollen.

☒ Auswahl  

Auswahl Suche

☒ Autoren  
☒ Ergebnisse der Datenprüfung  

Aus...



# Novo Brdo







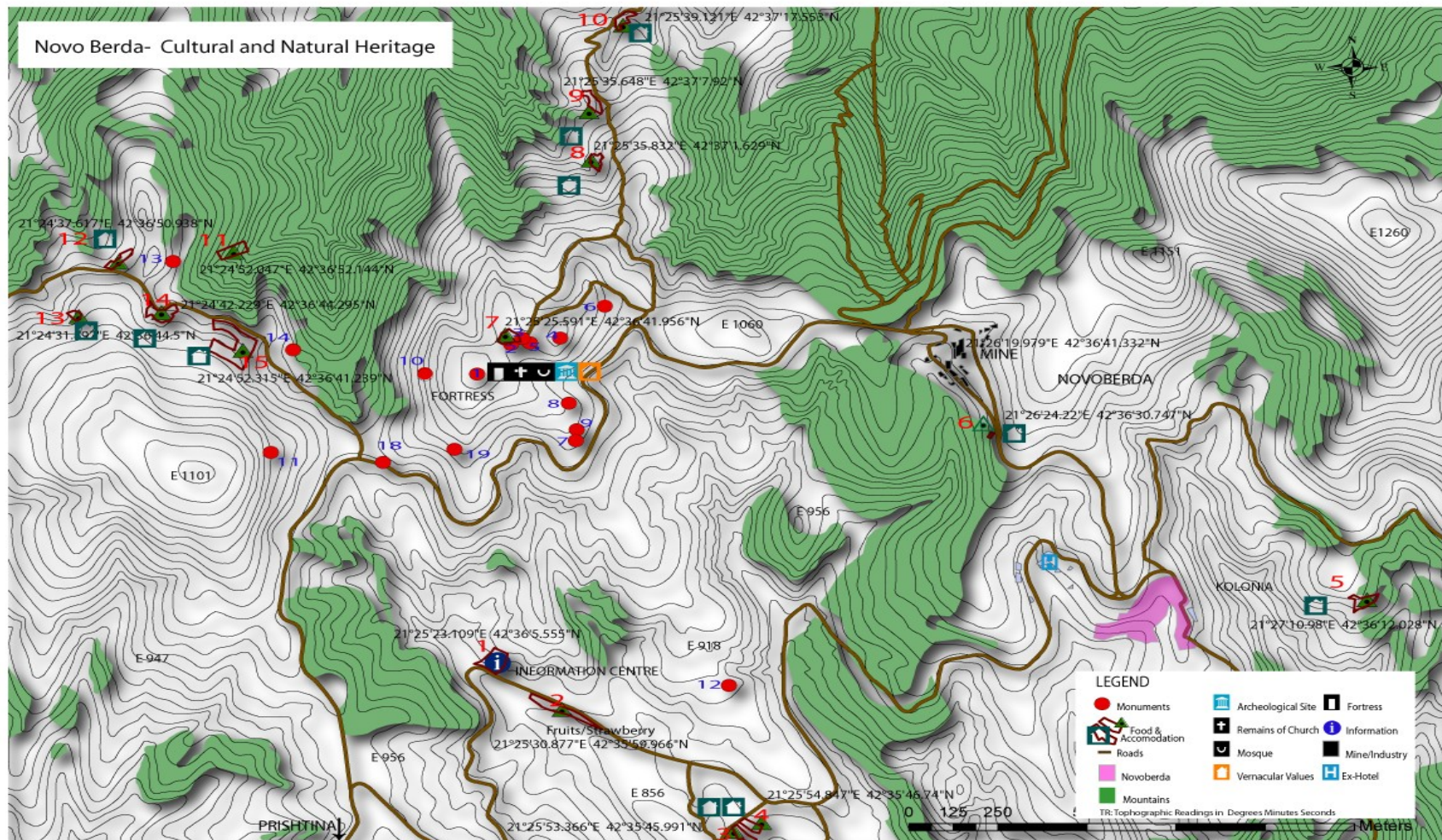


|           |          |                 |
|-----------|----------|-----------------|
| 21 Sep 10 | 7:05 pm  | Lulzim          |
| 29 Aug 10 | 5:36 pm  | user_5359       |
| 28 Jun 10 | 7:28 am  | h4ck3rm1k3      |
| 26 Jun 10 | 9:55 pm  | xybot           |
| 26 Jun 10 | 8:48 am  | FlosskPrishtina |
| 25 Jun 10 | 3:11 pm  | FlosskPrishtina |
| 25 Jun 10 | 1:07 pm  | Lulzim          |
| 25 Jun 10 | 10:07 am | FlosskPrishtina |
| 13 Jun 10 | 5:30 pm  | inthenameofmine |
| 12 Jun 10 | 9:17 am  | inthenameofmine |
| 13 Mar 10 | 11:04 am | bmog            |
| 13 Feb 10 | 8:55 am  | telegnom        |
| 9 Feb 10  | 9:02 pm  | telegnom        |





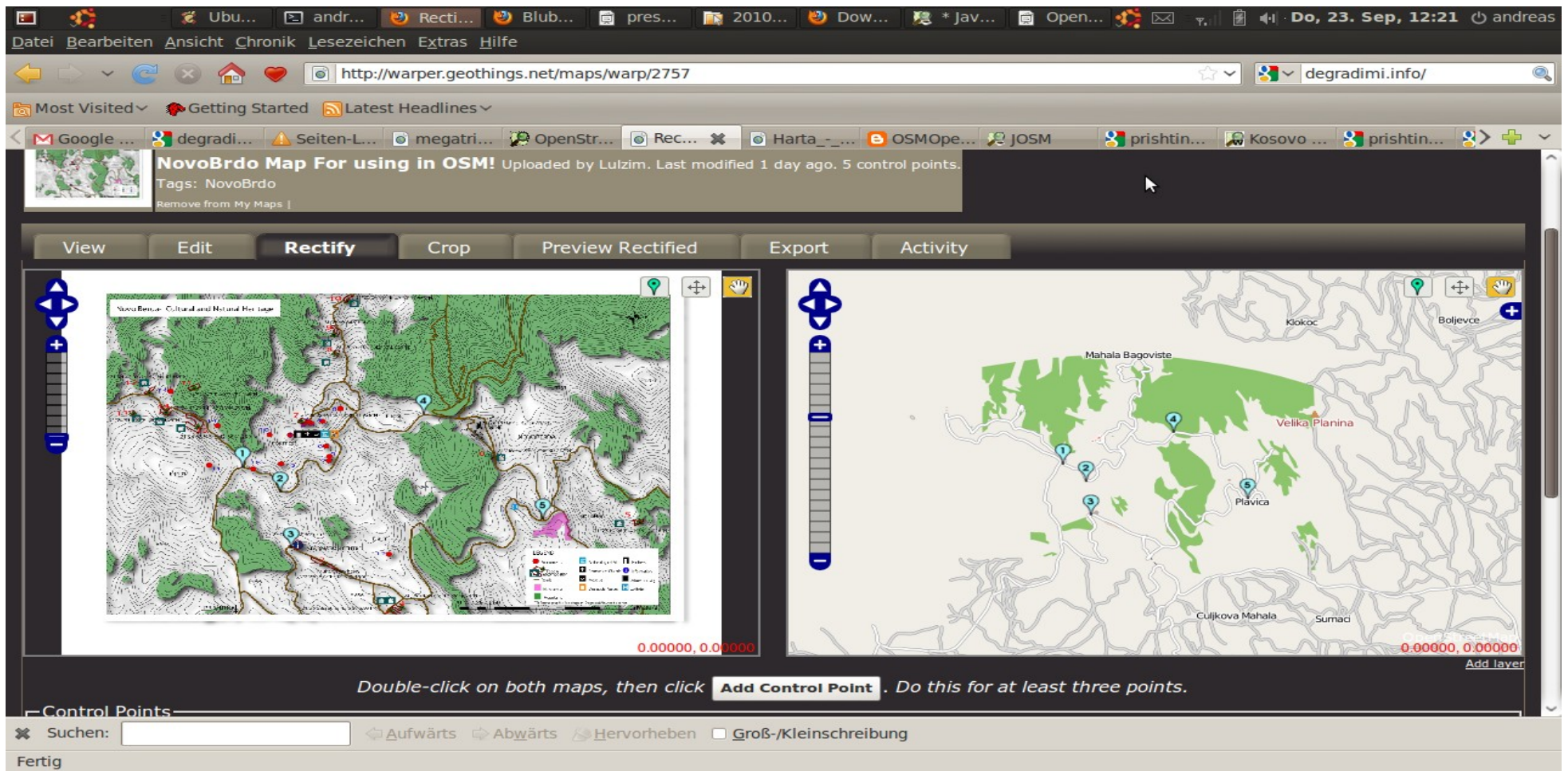
# original maps





# Warping / rectifying

- <http://warper.geothings.net/maps/warp/2757>



# Warping / rectifying

Ubu... andr... Warp... Blub... pres... 2010... Dow... \* Jav... Open... Do, 23. Sep, 12:24 andreas

Datei Bearbeiten Ansicht Chronik Lesezeichen Extras Hilfe

http://warper.geothings.net/maps/preview/2757 degradimi.info cad file

Most Visited Getting Started Latest Headlines

Google degradimi Seiten-L megatri OpenStr War Harta OSMOpe JOSM prishtin Kosovo prishtin

View Edit Rectify Crop **Preview Rectified** Export Activity

Viewing warped map

Suchen: Aufwärts Abwärts Hervorheben Groß-/Kleinschreibung

http://warper.geothings.net/maps/preview/2757#

# gjilan

[http://wiki.openstreetmap.org/wiki/WikiProject\\_Gjilan](http://wiki.openstreetmap.org/wiki/WikiProject_Gjilan)

# prizren

Data donations from city government

<http://warper.geothings.net/maps/2367>



# Dragash/Brod, unhabitat

[http://wiki.openstreetmap.org/wiki/WikiProject\\_Brod%28Kosovo%28](http://wiki.openstreetmap.org/wiki/WikiProject_Brod%28Kosovo%28)



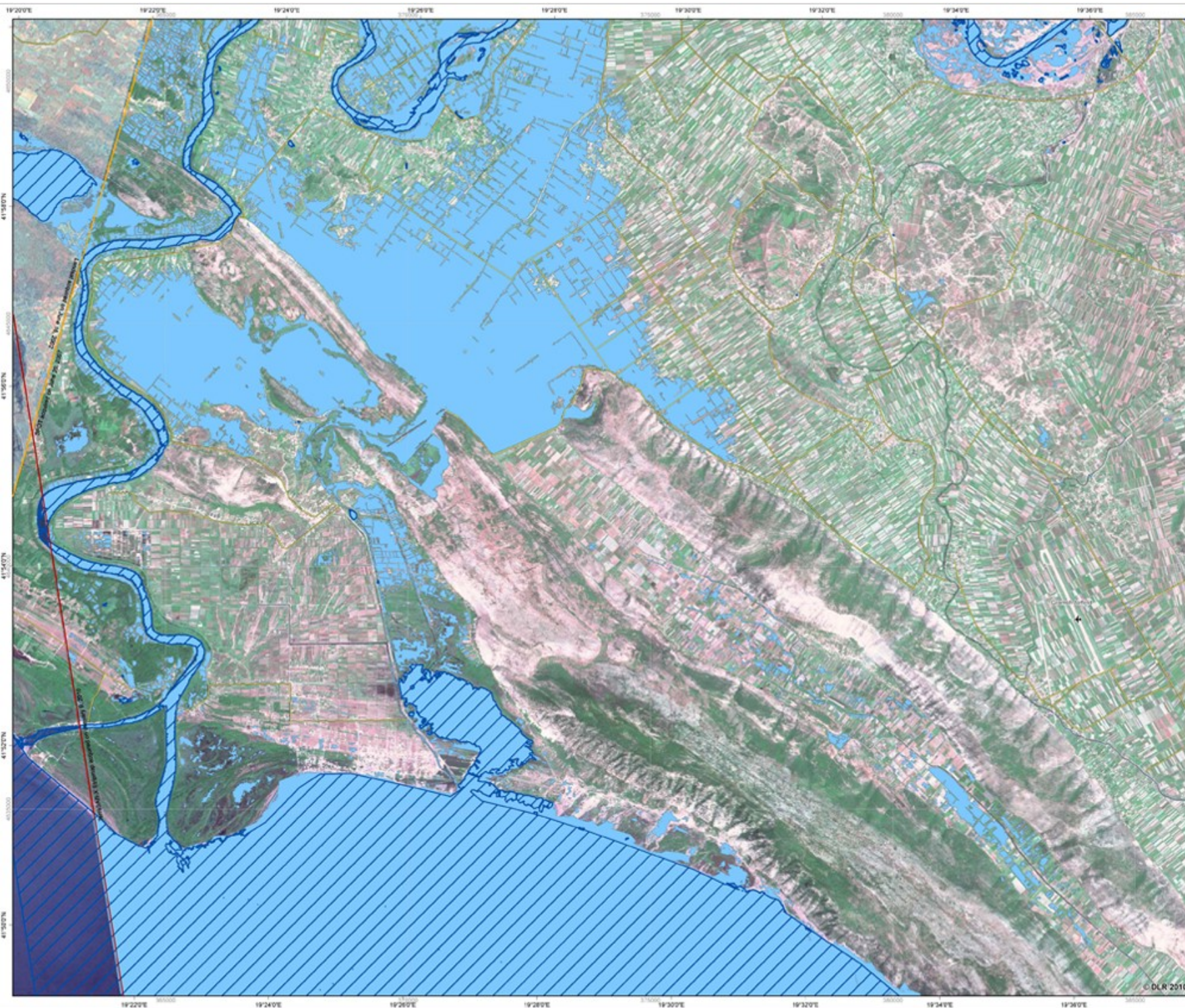
# Shkoder Albania

[http://wiki.openstreetmap.org/wiki/WikiProject\\_Shkoder](http://wiki.openstreetmap.org/wiki/WikiProject_Shkoder)

HAITI :

<http://vimeo.com/9182869>





Charter Call ID 284  
Glide No FL-2009-00266-ALB  
Product n°93

## ALBANIA - South of Shkoder

### Flood extent based on TerraSAR-X

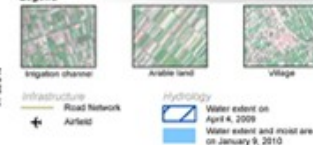
#### Detail Map

Scale 1:35,000

#### Location Diagrams



#### Legend



#### Extents of satellite scenes



#### Interpretation

Heavy rainfalls and unusually warm weather have caused flooding in the northwestern regions of Albania since January 4, 2010. The map shows the pre- and post-event water extents (as detected on April 4, 2010 and on January 9, 2010) around Shkoder. The analysis was based on TerraSAR-X Stripmap scenes. As backdrop a SPOT image acquired on June 20, 2007 and a Landsat image of June 16, 2002 were used. Streets have been digitized on the basis of the SPOT image and are only an extraction of the road network.

#### Cartographic Information



Local projection: UTM Zone 34 North, Datum: WGS 84  
Geographic projection: Lat/Lon (GMS), Datum: WGS 84

Scale: 1:35,000 for A1 prints

#### Data Sources

TerraSAR-X © German Aerospace Center (DLR) 2010  
SPOT 5 © CNES 2007  
Landsat-7 ETM © USGS 2002  
Vector data © OpenStreetMap 2010

#### Framework

The products elaborated for this Rapid Mapping Activity are realized to the best of our ability, within a very short time frame, during a crisis, optimizing the material available. All geographic information has limitations due to the scale, resolution, date and interpretation of the original source materials. No liability concerning the content or the use thereof is assumed by the producer.

The ZIG crisis maps are constantly updated. Please make sure to visit <http://www.zig.dlr.de> for the latest version of this product.

Map produced on January 10, 2010 by ZIG  
© DLR 2010

[zig@dlr.de](mailto:zig@dlr.de)  
<http://www.zig.dlr.de>



Center for Satellite Based Crisis Information  
- Emergency Mapping & Disaster Monitoring -  
a service of DLR





# Shkoder Hotels



**OpenStreetMap**

The Free Wiki World Map

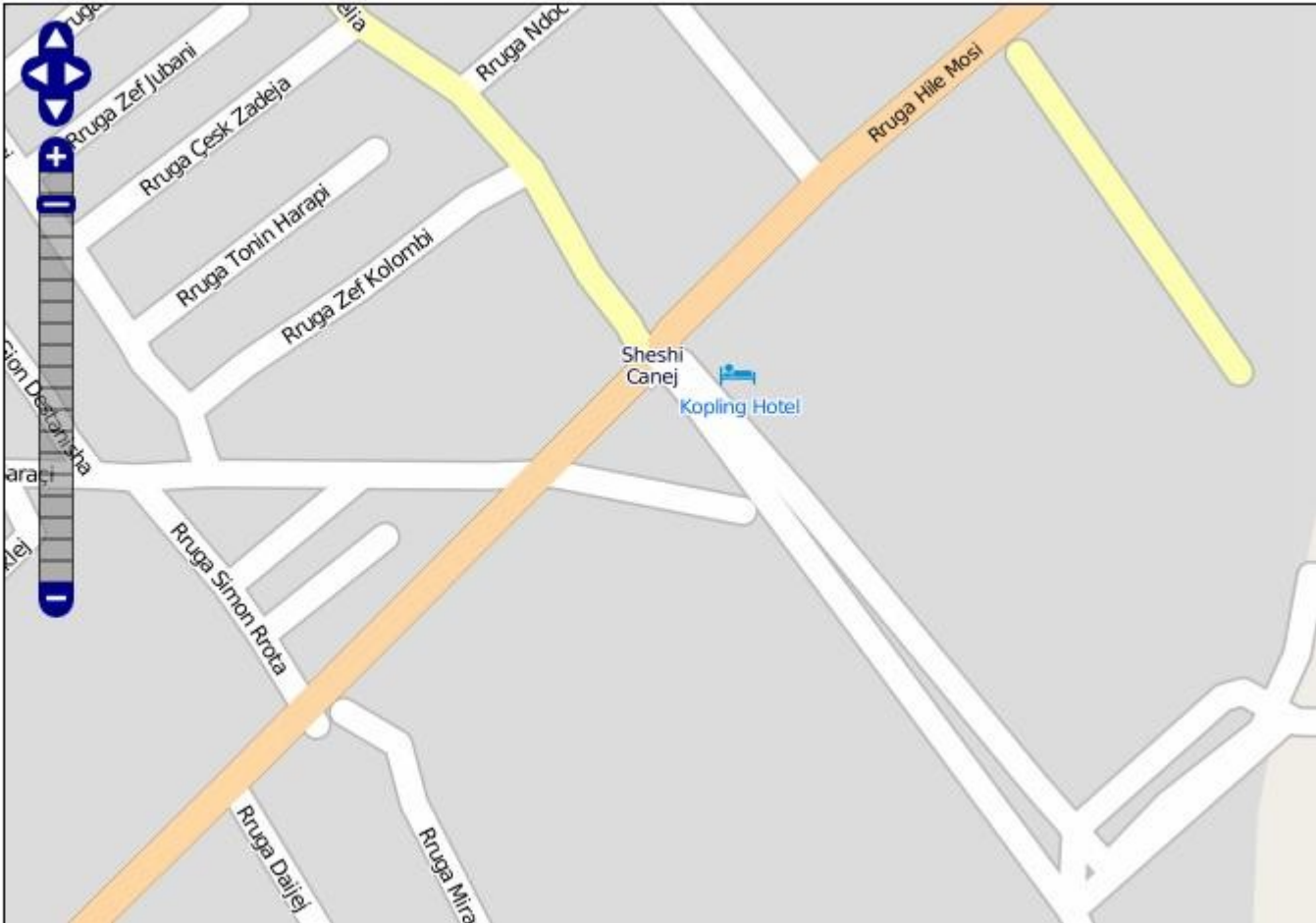
OpenStreetMap is a free editable map of the whole world. It is made by people like you.

OpenStreetMap allows you to view, edit and use geographical data in a collaborative way from anywhere on Earth.

OpenStreetMap's hosting is kindly supported by the [UCL VR Centre](#) and [bytemark](#). Other supporters of the project are listed in the [wiki](#).

**Help & Wiki**  
**Copyright & License**  
**News blog**  
**Shop**

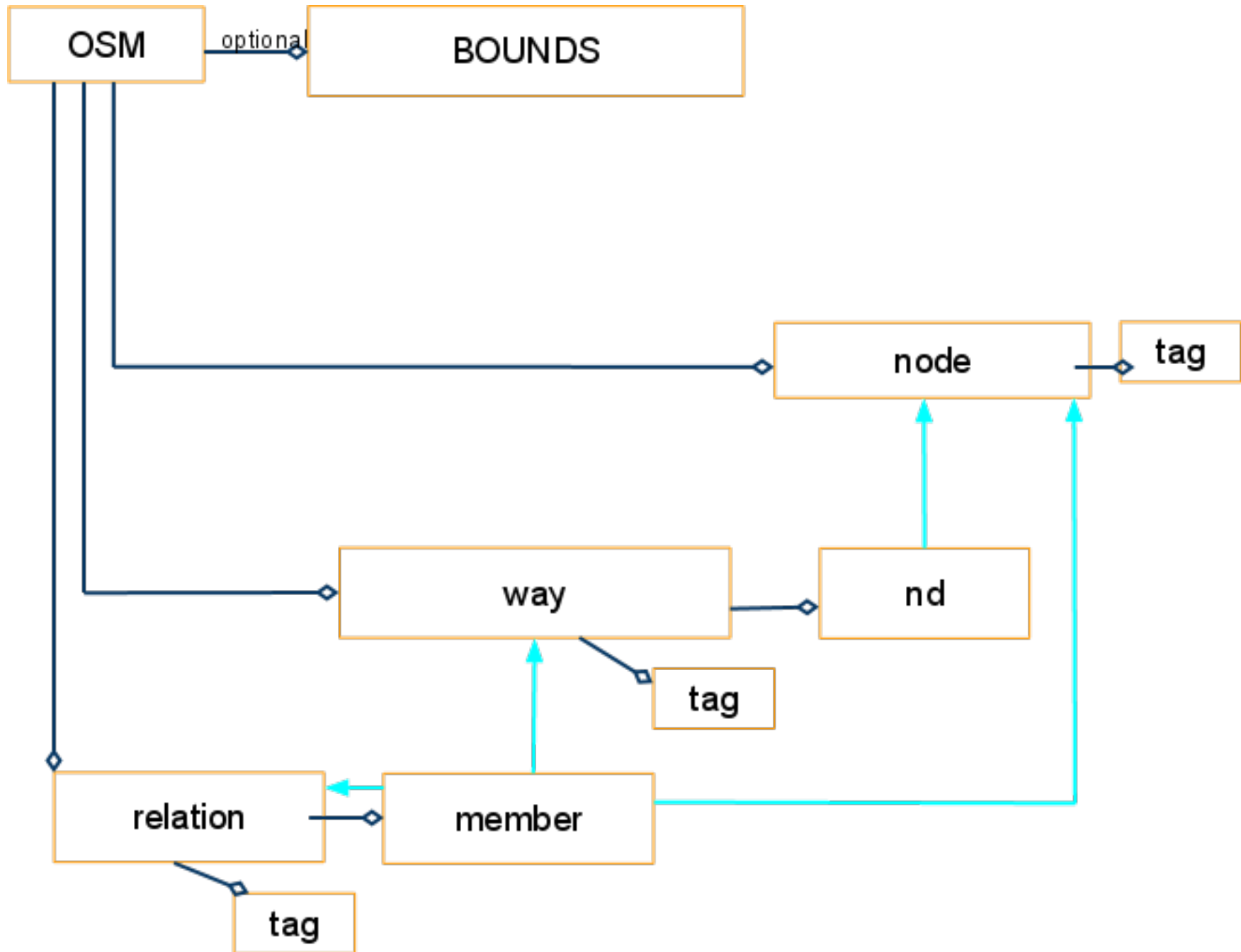
**View** **Edit** **History** **Export** **GPS Traces** **User Diaries**



The map displays a street network in Shkoder. Labeled streets include Rruga Zef Jubani, Rruga Çesk Zadeja, Rruga Tonin Harapi, Rruga Zef Kolombi, Rruga Ndoc, Rruga Hile Mosi, Rruga Simon Rota, Rruga Daljel, and Rruga Mira. A blue icon representing a hotel is labeled 'Kopling Hotel' and is located near the intersection of Rruga Zef Kolombi and Rruga Hile Mosi. The text 'Sheshi Canej' is also visible near the hotel icon. On the left side of the map, there is a vertical scale bar and a compass rose.



# OpenStreetMap XML format : .osm





# OSM file format

The OSM xml file format is a simple, flat xml file that represents the graph of the world.

The root OSM node contains a bounds object that describes the area in the file. Then the file then has a sequence of node, way and relation objects each with a set of tags and some sub objects. Ways reference nodes, and relations can reference ways, nodes or other relations.

See also :

[http://wiki.openstreetmap.org/wiki/Data\\_Primitives](http://wiki.openstreetmap.org/wiki/Data_Primitives) and  
<http://wiki.openstreetmap.org/wiki/.osm>

# Tiles

Images are not considered dangerous to javascript.  
You cannot read xml files from another server due to xss restrictions.  
Images can be loaded from other servers with no problem.

<http://osmopenlayers.blogspot.com/2010/06/test-of-external-svg.html>

Tiles and tile rendering

There is an ongoing process of fetching map data via the API, and rendering maps as [raster images](#) known as tiles. Many new tiles need to be rendered to achieve full (global) coverage, but also updates to the underlying map data will mean that existing tiles need to be re-rendered.

# Rendering

Marking a tile dirty **does not** mark all sub tiles as dirty. If you get 'More OpenStreetMap coming soon...' on a tile, it means there was no data for that tile and it is now in the queue to be rendered. You can find a tile status by getting a tile URL (right click and 'get URL for image' or similar):

- <http://tile.openstreetmap.org/7/63/42.png>

and then add /status on the end:

- <http://tile.openstreetmap.org/7/63/42.png/status>

which will tell you its creation timestamp and dirty status.

If you want to make a tile render before the seven day expiry then you can mark it as dirty by appending /dirty:

- <http://tile.openstreetmap.org/7/63/42.png/dirty>

# Slippy Map

See [Slippy Map](#). The main map appearing on the [openstreetmap.org](#) homepage is javascript interface letting you zoom and pan ("slippy" draggable panning). What happens is, the website (the 'index' view of the rails app) invokes [OpenLayers](#). As it does so, it passes in a latitude and longitude based on the users last viewed location or URL params. OpenLayers does its client side javascript magic, to figure out which 'tile' images to fetch from the tile server.

# Your own slippy map :

[http://wiki.openstreetmap.org/wiki/Deploying\\_your\\_own\\_Slippy\\_Map](http://wiki.openstreetmap.org/wiki/Deploying_your_own_Slippy_Map)

Export Embeddable HTML:

<http://osmopenlayers.blogspot.com/2010/06/test-of-embedded-osm.html>

open layers is the most common usage : [OpenLayers](#) ([openlayers.org](http://openlayers.org)) is an feature-rich free open source JavaScript library.

[Mapstraction](#) ([mapstraction.com](http://mapstraction.com)) is an open source javascript wrapper around the above libraries (and several others) allowing you to swap your choice of map library provider without re-coding anything.



# tiles@home

<http://wiki.openstreetmap.org/wiki/Tiles@home>

[Osmarender](#) - XSLTs which does OSM XML to SVG transformation. [tiles@home](#) is a project which uses Osmarender to create a tile set for server a map layer alongside Mapnik's, but Osmarender can also be a good option for doing one-off renderings. This is the easiest way of getting maps in SVG format without any data dropped for better display, which allows you to do post-processing tidy up tweaks.

T@H has a server software, Tahngo (generation 2), running at the [Tiles@home website](#), which get requests to render tiles from updated mapdata. There are [many people](#) who run the client software on their computers that ask what map-tile to render and contribute their results back to the server.

See also <http://www.informationfreeway.org/>

# The world : Planet.osm

The world file, planet.osm, is an OSM XML File

<http://planet.openstreetmap.org/>

<http://wiki.openstreetmap.org/wiki/Planet.osm>

Osmosis can process the Planet.osm file to break it down into manageable chunks and to create/apply patches:

<http://wiki.openstreetmap.org/wiki/Osmosis>

# Planet Mirrors

<http://wiki.openstreetmap.org/wiki/Planet.osm#Mirrors>

eg :

<http://hypercube.telascience.org/planet/>

extracts :

<http://download.geofabrik.de/osm/>

<http://downloads.cloudmade.com/>

# An der Sandelmühle 35

<http://www.openstreetmap.org/?lat=50.162994&lon=8.650918&zoom=18&layers=B000FTF>



# Example Node

```
<node id="34157406" lat="50.163021" lon="8.6511725"  
user="MichaH" uid="8464" visible="true" version="3"  
changeset="1841979" timestamp="2009-07-16T07:21:32Z">  
<tag k="amenity" v="restaurant"/>  
<tag k="name" v="Sandelmühle"/>  
</node>
```

<http://www.openstreetmap.org/browse/node/34157406>



# Example Way Building

```
<way id="30616450" user="MichaH" uid="8464" visible="true" version="2"
changeset="1841979" timestamp="2009-07-16T07:21:45Z">
<nd ref="338531695"/><nd ref="338531697"/><nd ref="338531700"/>
<nd ref="338531701"/><nd ref="338531703"/>
<nd ref="338531704"/><nd ref="338531705"/>
<nd ref="338531706"/><nd ref="338531702"/>
<nd ref="338531699"/><nd ref="338531698"/>
<nd ref="338531696"/><nd ref="338531695"/>
<tag k="addr:city" v="Frankfurt"/>
<tag k="addr:housenumber" v="35"/>
<tag k="addr:postcode" v="60439"/>
<tag k="addr:street" v="An der Sandelmühle"/>
<tag k="building" v="yes"/>
</way>
```

<http://www.openstreetmap.org/browse/way/30616450>

# Example Relation : Urselbach Mühlenwanderweg

<http://www.openstreetmap.org/browse/relation/78067>

```
<relation id="78067" user="PHerison" uid="28378" visible="true" version="8" changeset="3304901" timestamp="2009-12-06T12:50:33Z">
<member type="node" ref="352783704" role=""/>
<member type="way" ref="4730851" role=""/>
.....
<member type="way" ref="23301887" role=""/>
<member type="way" ref="4913286" role=""/>
<tag k="name" v="Urselbach Mühlenwanderweg"/>
<tag k="network" v="lwn"/>
<tag k="operator" v="Stadt Oberursel"/>
<tag k="osmc:symbol" v="blue:white: :UM:yellow"/>
<tag k="route" v="foot"/>
<tag k="symbol" v="stilisiertes Mühlrad, Wasserlauf und Wasserflohkrebs"/>
<tag k="type" v="route"/>
</relation>
```

# Example Way : An der Sandelmühle

<http://www.openstreetmap.org/browse/way/4730848>

<http://www.openstreetmap.org/api/0.6/way/4730848>

```
<way id="4730848" visible="true" timestamp="2010-05-30T07:57:35Z" version="9" changeset="4849228" user="baeuchle"
uid="221387">
<nd ref="29536069"/><nd ref="30149135"/><nd ref="338531293"/><nd ref="30149137"/>
<nd ref="30149138"/><nd ref="338531300"/><nd ref="30149140"/><nd ref="440107927"/>
<nd ref="382390507"/><nd ref="440107939"/><nd ref="30149142"/>
<nd ref="30149143"/><nd ref="30149145"/><nd ref="30149146"/><nd ref="30149147"/>
<nd ref="30149148"/><nd ref="441278967"/><nd ref="30149149"/>
<nd ref="30149150"/><nd ref="30933456"/><nd ref="758595966"/>
<nd ref="43539957"/><nd ref="561211"/>
<tag k="highway" v="residential"/>
<tag k="name" v="An der Sandelmühle"/>
</way>
```

# Osmosis

<http://wiki.openstreetmap.org/wiki/Osmosis>

Osmosis is the main tool for OSM processing. It can read and write the osm file, read database, filter and split and merge areas. It is written in java and runs on the command line.

Importing data, extracting boxes, exporting data.

# OSMosis extract box

[http://wiki.openstreetmap.org/wiki/Osmosis#Extracting\\_bounding\\_boxes](http://wiki.openstreetmap.org/wiki/Osmosis#Extracting_bounding_boxes)

```
bzcat downloaded.osm.bz2 | osmosis\  
--read-xml enableDateParsing=no file=/dev/stdin\  
--bounding-box top=49.5138 left=10.9351 bottom=49.3866  
right=11.201 --write-xml file=-\  
| bzip2 > extracted.osm.bz2
```



# Potlatch

[Potlatch](#) is the online editor appearing on the 'edit' tab. Again the rails app has a view for the 'edit' tab. Potlatch is a flash object embedded on the web page. It is written in ActionScript. While running in the browser it makes calls to a special part of the API called the 'AMF Controller'. Because of browser security constraints it cannot be embedded on any website other than openstreetmap.org (since it must make HTTP requests to the API)

# Mapnik

[Mapnik](#) is a rendering system which powers the display which is currently the [Slippy Map](#) default. The rendering process runs on the 'tile' server, and Mapnik tile images are served from that machine. This renderer takes its data from a postgres database (also on the tile server). This database holds data in the format expected by Mapnik, using [PostGIS](#) extensions. This is an entirely different format to the database used on the core OSM database server. The mapnik database is populated by occasionally running an [osm2pgsql](#) script on the weekly [Planet.osm](#) dump. Since Feb 2009 it is also being populated more regularly with the application of [Osmosis](#) diffs (hourly or even minutely)

# Mapnik installation

Test installation :

Generate Tile :

```
python generate_tiles.py  
    bbox = (17.96,39.34,24.1,43.35)  
    render_tiles(bbox, mapfile, tile_dir, 1, 18, "World")
```

Hacked version of main page.

<http://xhema.flossk.org:8080/mapdata/05/layerstest/OpenStreetMap.html>

Using mapfish/openlayers

<http://xhema.flossk.org:5000/customtiles.html>

# OSM API

OSM uses a REST API (i.e. OSM XML over HTTP, with use of simple URLs for object access, and standard HTTP response codes. ) for interacting with the server.

[http://wiki.openstreetmap.org/wiki/API\\_v0.6](http://wiki.openstreetmap.org/wiki/API_v0.6)

It supports basic operations to read and write data to the server.

Before editing and data, users must authenticate against the server. All user changes are collected in a changeset, which is assigned a unique id.

The API logic is all part of the same Ruby on Rails application which powers the OSM front end website.

# Representational State Transfer API

- **Read data :**
- **Get map data :** GET /api/0.6/map?bbox=*left,bottom,right,top*
- **get changeset** GET /api/0.6/changeset/#id
- **Authenticated Data :**
- **Create Changeset:** PUT /api/0.6/changeset/create
- **Close Changeset** Close: PUT /api/0.6/changeset/#id/close
- **create data :** Create: PUT /api/0.6/[node|way|relation]/create
- **Update data :** PUT /api/0.6/[node|way|relation]/#id



# drupal

- Some form of user feedback is needed, we need a facebook app for openstreetmap, a drupal cms with forms made for all major datatypes, including pictures

# stalled : porting of mapzen code to haxe

<http://gitorious.org/mapzen-port/mapzen-port>

porting of mapzen code to haxe. this should be reworked to  
port to java

# osm2blender 3d model

[http://bazaar.launchpad.net/~kosova/  
%2Bjunk/openstreetmapkosova/annotate/head  
%3A/kmz\\_ImportWithMesh.py](http://bazaar.launchpad.net/~kosova/%2Bjunk/openstreetmapkosova/annotate/head%3A/kmz_ImportWithMesh.py)

[http://permalink.gmane.org/gmane.comp.gis.openstreetmap.de  
vel/19013](http://permalink.gmane.org/gmane.comp.gis.openstreetmap.devel/19013)

<http://www.youtube.com/watch?v=Hw9aW3il5fs>

# patches for osm2pgsql

<http://github.com/h4ck3rm1k3/osm2pgsql>

# LibreDWG compiled with c++

[http://github.com/h4ck3rm1k3/Libre  
DWGCPlusPlus](http://github.com/h4ck3rm1k3/LibreDWGCPlusPlus)



geocoder (usa) tool,  
autotoolized

<http://github.com/h4ck3rm1k3/AutoToolsGeocoder/tree/master/geocoder/>

# autotoolize of mapnik and also reverse engineering

<http://github.com/h4ck3rm1k3/MapNickAutotools>

# My first scripts for osm

<http://bazaar.launchpad.net/~kosova/+junk/openstreetmapkosova/files>

tons of my first tools to process openstreetmap data.

- forest walker to trace green data from landsat
- <http://bazaar.launchpad.net/~kosova/+junk/openstreetmapkosova/files/head:/forestwalker/>
- duplicate removal tools

# Example kpaonline import

import tool for kpaonline data

[How To](#) |  
[scraper](#)

Extracting data from html, Projecting it to lat/lon processing it into osm.

1. use wget to get all the items into a directory :  
each town is in kpaonline in a view from here:

<http://www.kpaonline.org/AdminMun.asp?mun=Pristina>

wget -r -l2 -d<http://www.kpaonline.org/>

<http://www.kpaonline.org/AdminMun.asp?mun=Pristina>

# reverse engineering of the DirectDWG code

<http://github.com/h4ck3rm1k3/InDirectDWG>

# patches for shp2osm

<http://github.com/h4ck3rm1k3/shp2osm>



twonickels with an dxf2osm  
tool

<http://github.com/h4ck3rm1k3/TwoNickels>

# wikipedia scanner with openstreetmap point collector

[https://code.launchpad.net/~jamesmikedupont/  
+junk/openstreetmap-wikipedia](https://code.launchpad.net/~jamesmikedupont/+junk/openstreetmap-wikipedia)

# advanced c++ template processing of openstreetmap data.

<https://code.launchpad.net/~jamesmikedupont/+junk/EPANatReg>

patch for inkscape to include  
openstreetmap photomapping  
feature

<https://code.launchpad.net/~jamesmikedupont/inkscape/osm-inkscape>

# storage and splitting of osm files for git

<http://gitorious.org/osmgit/osmgit-test>

Torrent files for these:

<http://osmopenlayers.blogspot.com/2010/06/first-running-of-osm-gittorrent.html>

# address database extractor for aspx database webpages

<https://code.launchpad.net/~jamesmikedupont/aspxezxs/aspxtraktor>



mapfish python

# osm task

## OSM Task:

We have a new tool to help mark  
the areas being worked on.

server is installed here, please  
check it out

<http://xhema.flossk.org:8881/>

OSMTask is described here in brief.

[http://wiki.crisiscommons.org/wiki/  
Camp\\_Roberts\\_Planning#OSM\\_  
Grid-](http://wiki.crisiscommons.org/wiki/Camp_Roberts_Planning#OSM_Grid-Square_Checkout_Server_.28GC_S.29_.28job_tasking.29)

[Square\\_Checkout\\_Server\\_.28GC  
S.29\\_.28job\\_tasking.29](http://wiki.crisiscommons.org/wiki/Camp_Roberts_Planning#OSM_Grid-Square_Checkout_Server_.28GC_S.29_.28job_tasking.29)

# sahana

<http://xhema.flossk.org:8000/sahana/default/index> sahana

# converting kml markers to osm

[http://gitorious.org/macedonian\\_pois/macedonian\\_pois](http://gitorious.org/macedonian_pois/macedonian_pois)

# my mapserver files

<http://gitorious.org/albanianfloodingcrisiscamp>

# gpsbabel to covert data

<http://www.gpsbabel.org/>

<http://wiki.openstreetmap.org/wiki/GPSBabel>



# mapserver

<http://mapserver.org>

# qgis

<http://www.qgis.org/>

# map warper to overly photos

<http://warper.geothings.net/>

# road matcher

<http://wiki.openstreetmap.org/wiki/Roadmatcher>

# contour finding, cutting ways

<http://osmopenlayers.blogspot.com/2010/03/first-version-of-way-cutter.html>

# gwt osm , google web toolkit osm

<http://wiki.github.com/h4ck3rm1k3/GWTOsm/>

<http://wiki.openstreetmap.org/wiki/JOSM/GwtOsm>

port of josm to gwt.

<http://xhema.flossk.org:8180/GWTOSM/>

<http://osmopenlayers.blogspot.com/2010/06/gwtosm-example-output.html>



# STALLED porting josm to gjc

<http://github.com/h4ck3rm1k3/josm>

The porting of josm to gjc is an interesting project to help check the gnu compiler for java and the classpath.

Right now on hold for working on GWTOSM.

# stalled : open address db

[Controller/Geo/GeoNames.pm](#)

here is a simple perl catalyst controller that does a geo names  
lookup and displays a openlayers map of the location found.  
uses the GEONames Model

[Model/Geo/GeoNames.pm](#)

stalled :traindirector would be great if it could read osm files.

<http://gitorious.org/osm-traindirector/debian-packages>

I just started on the debian packaging, did not look into more details.

# kosova.wikia.org

- Wikipages with various sources of names of streets
- Wikipages for streets

<http://kosova.wikia.org>